A M E N D M E N T S   T O   T H E   C L A I M S

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Currently Amended) A method, comprising:

determining via a plurality of flags stored at a PCI device that the PCI device is to perform a first function ~~a first software component is to use a device~~, the PCI device ~~device~~ being shared with a first software component associated with the first function and a second software component associated with a second function, wherein the plurality of flags comprise a first component flag to indicate free or busy, a second component flag to indicate free or busy, and a turn flag to indicate the first component or the second component;

setting the first component flag to busy;

setting the turn flag to first component; and

~~; and~~

performing the first function ~~arranging via information stored at the device to use the device~~ when the second software component is not using the PCI device,

wherein the first function may access the PCI device if either the second component flag is set to free or if the turn flag is set to first component.

2. (Currently Amended) The method of claim 1, further comprising

waiting until either the first component flag is set to free or the turn flag is set to second component before using the device; and

performing the second function when the first software component is not using the PCI device.

~~wherein the information stored at the device includes:~~

3

a first component flag,

a second component flag, and

a turn flag,

wherein the first and second component flags can indicate free or busy and the turn flag can indicate first component or second component.

3. (Original) The method of claim 2, wherein the first and second component flags are initialized to free.

4. (Currently Amended) The method of claim 3, wherein said arranging further comprises further comprising:

setting the first component flag to busy; and

setting the turn flag to second component.

5. (Currently Amended) The method of claim 4, wherein said arranging further comprises further comprising:

waiting until either the second component flag is free or the turn flag is first component before using the device.

6. (Currently Amended) The method of claim 5, wherein said arranging further comprises further comprising:

setting the first component flag to free after using the device.

7. (Currently Amended) The method of claim 1, wherein said ~~arranging~~ performing is ~~performed~~ in accordance with at least one of: (i) Dekker's algorithm, and (ii) Peterson's algorithm.

8. (Currently Amended) The method of claim 1, wherein the ~~information~~ plurality of flags are ~~is~~ stored using pre-fetchable storage registers at the device.

9. (Original) The method of claim 1, wherein at least one of the first and second software components comprise a device driver.

10. (Currently Amended) The method of claim 1, wherein the PCI device comprises at least one of: (i) a network adapter, and (ii) a disk controller.

11. (Currently Amended) The method of claim 1, wherein the PCI device comprises a network controller, the first software component comprises a network driver, and the second software component comprises an encryption driver.

12. (Original) The method of claim 1, wherein the first and second software components access the device via at least one of (i) a peripheral component interconnect bus and (ii) a universal serial bus.

13. (Currently Amended) An apparatus, comprising:

a storage medium having stored thereon instructions that when executed by a machine result in the following:

determining via a plurality of flags stored at a PCI device that the PCI device is to perform a first function , the PCI device being shared with a first software component associated with the first function and a second software component associated with a second function, wherein the plurality of flags comprise a first component flag to indicate free or busy, a second component flag to indicate free or busy, and a turn flag to indicate the first component or the second component;

setting the first component flag to busy;

setting the turn flag to first component; and

performing the first function when the second software component is not using the PCI device,

wherein the first function may access the PCI device if either the second component flag is set to free or if the turn flag is set to first component.

~~determining that a first software component is to use a device, the device being shared with a second software component, and~~

~~arranging via information stored at the device to use the device when the second software component is not using the device.~~

14. (Currently Amended) The apparatus of claim 13, ~~wherein the information stored at the device comprises~~ wherein the instructions further result in:

waiting until either the first component flag is set to free or the turn flag is set to second component before using the device; and

performing the second function when the first software component is not using the PCI device.~~a first component flag,~~

~~a second component flag, and~~

6

~~a turn flag,~~

~~wherein the first and second component flags can indicate free or busy and the turn flag can indicate first component or second component.~~

15. (Original) The apparatus of claim 14, wherein the first and second component flags are initialized to free.

16. (Currently Amended) The apparatus of claim 15, wherein ~~s~~the instructions further result in:~~aid arranging further comprises~~:

        setting the first component flag to busy, and

        setting the turn flag to second component.

17. (Currently Amended) The apparatus of claim 16, wherein the instructions further result in~~said arranging further comprises~~:

        waiting until either the second component flag is free or the turn flag is first component one before using the device.

18. (Currently Amended) The apparatus of claim 17, wherein the instructions further result in~~said arranging further comprises~~:

        setting the first component flag to free after using the device.

19. (Currently Amended) The apparatus of claim 13, wherein said ~~arranging~~ performing is ~~performed~~ in accordance with at least one of: (i) Dekker's algorithm, and (ii) Peterson's algorithm.

20. (Currently Amended) The apparatus of claim 13, wherein the ~~information~~ plurality of registers ~~is~~ are stored using pre-fetchable storage registers at the device.

21. (Original) The apparatus of claim 13, wherein at least one of the first and second software components comprise a device driver.

22. (Currently Amended) The apparatus of claim 13, wherein the PCI device comprises at least one of: (i) a network adapter, and (ii) a disk controller.

23. (Currently Amended) An apparatus, comprising:

a processor; ~~and~~

a bus interface, and

a storage medium having stored thereon instructions that when executed by the processor result in the following:

determining via a plurality of flags stored at a PCI device that the PCI device is to perform a first function , the PCI device being shared with a first software component associated with the first function and a second software component associated with a second function, wherein the plurality of flags comprise a first component flag to indicate free or busy, a second component flag to indicate free or busy, and a turn flag to indicate the first component or the second component;

setting the first component flag to busy;

setting the turn flag to first component; and

performing the first function when the second software component is not using the PCI device,

wherein the first function may access the PCI device if either the second component flag is set to free or if the turn flag is set to first component,

wherein ~~information stored at a device~~the plurality of flags are accessed via the bus interface ~~is to arrange for a first software component of the processor to use the device when a second software component of the processor is not using the device~~.

24. (Original) The apparatus of claim 23, wherein the device comprises a peripheral device and the information is stored using pre-fetchable registers at the peripheral device.

25. (Currently Amended) A system, comprising:

a processor;

a network adapter; ~~and~~

an Ethernet port coupled to the network adapter; and

a storage medium having stored thereon instructions that when executed by the processor result in the following:

determining via a plurality of flags stored at a PCI device that the PCI device is to perform a first function , the PCI device being shared with a first software component associated with the first function and a second software component associated with a second function, wherein the plurality of flags comprise a first component flag to indicate free or busy, a second

component flag to indicate free or busy, and a turn flag to indicate the first component or the second component;

setting the first component flag to busy;

setting the turn flag to first component; and

performing the first function when the second software component is not using the PCI device,

wherein the first function may access the PCI device if either the second component flag is set to free or if the turn flag is set to first component.

~

wherein information stored at the network adapter is to arrange for a first software component of the processor to use the network adapter when a second software component of the processor is not using the network adapter.

26. (Original) The system of claim 25, wherein the first software component comprises a network driver and the second software component comprises an encryption driver.